20.5 The Multi-threaded Parity-Protected 128-Word Register Files on a Dual-Core Itanium®-Family Processor

10:45 AM

L. Wang[1], E. Fetzer[2], J. Jones[3]
[1]University of Connecticut, Storrs, Connecticut
[2]Hewlett Packard, Fort Collins, CO
[3]Intel, Fort Collins, CO

The dual-thread 18-port 128wx82b FPU register file, and the 22-port 128wx65b integer register file of the microprocessor is described. Parity embedded into each register provides soft error detection. The design integrates a charge-compensated thread switch and power-saving features to operate at 1.1V consuming 400mW at maximum frequency.

20.6 A 32b 64-Word 9-Read-Port/7-Write-Port Pseudo Dual-Bank Register File Using Copied Memory Cells for a Multi-Threaded Processor

11:15 AM

M. Sumita, Y. Ikeda
Matsushita, Kyoto, Japan

A method for copying memory cells to reduce the size of the register file for a multi-threaded processor is proposed. The number of transistors in the memory cell is reduced to 70% and the total bit and word lines is reduced to 63%. The register file is implemented in a 100nm CMOS process. The size of the register file is reduced to 62% of a conventional register file. The power consumption is reduced by 50%.

20.7 A 3Gb/s/ch Transceiver for RC-limited On-Chip Interconnects

11:45 AM

D. Schinkel[1], E. Mensink[1], E. Klumperink[1], E. van Tuijl[1,2], B. Nauta[1]
[1]University of Twente, Enschede, The Netherlands
[2]Philips, Eindhoven, The Netherlands

A bus-transceiver chip in 0.13μm CMOS uses 10mm uninterrupted differential interconnects of 0.8μm pitch (82MHz RC-limited bandwidth). The chip achieves 3Gb/s/ch using a pulse-width pre-emphasis technique in combination with resistive termination while twisted interconnects mitigate crosstalk. Power consumption is 6mW/ch at a 1.2V supply.

CONCLUSION 12:15 PM

20.5 The Multi-Threaded, Parity-Protected 128-Word Register Files on a Dual-Core Itanium®-Family Processor

Eric S. Fetzer[1], Lei Wang[2], Jared Jones[1]

[1]Intel, Fort Collins, CO
[2]University of Connecticut, Storrs, CT

Multi-threaded microprocessors require multiple sets of register files to process concurrent instruction streams. This significantly complicates the register-file (RF) design and exacerbates reliability problems. This paper describes the dual-threaded, 18-port (8-read, 10-write), 128word × 82b floating-point register file (FRF), and the 22-port (12-read, 10-write), 128 × 65b integer register file (IRF) of the processor-code named Montecito [1]. A memory circuit is designed that consists of two storage cells, each bit accessible by two different threads. A charge-compensation technique is introduced to mitigate charge-sharing noise induced by thread-switch events. The dual-threaded implementation essentially doubles the number of memory cells and the RF becomes even more susceptible to errors caused by high-energy particles. A low-complexity parity-checking scheme is embedded into each register to provide soft error detection. The current design also integrates several power-saving features to achieve energy-efficiency and reliable operation.

To support dual-threaded execution, each register bitcell incorporates two identical storage cells (Fig. 20.5.1). A control signal WRITEH is driven high during write operations, reducing the contention between the two cross-coupled inverters. Four transmission gates determine the thread selection, where each storage cell is exclusively accessible by either thread. Switching threads can induce charge-sharing noise between the storage cells b0/b1 and the internal bitlines ida/idb. Consider the worst-case scenario where the two storage cells contain different voltages. Each internal bitline will have the same voltage as the storage cell it is connected to. A thread switch is performed by flipping the thread signal. Each storage cell is then connected to the internal bitline having a different voltage value. Charge is redistributed momentarily between the storage cells and the internal bitlines. This may flip the storage cells and cause logic failures. To prevent this problem, two charge-compensation pFETs p0/p1 are introduced. The signal writel, connected to the sources of p0/p1, remains high during thread switch, thereby compensating for possible charge loss at b0/b1 through p0/p1 (Fig. 20.5.2). This technique does not induce additional contention during write operations when writel switches to virtual ground. Moreover, this technique improves write timing. A high voltage at the storage node (b0/b1) will be pulled down slightly by writel at the beginning of a write operation, making it easier to write a "0". For writing a "1", the charge-compensation pFETs p0/p1 help boost b0/b1 to VDD quickly at the end of a write when writel goes high.

The register files are relatively large because of the dual-threaded implementation. This compounded with the high-density layout makes the RFs susceptible to soft errors. Therefore, the RF data arrays integrate parity checking hardware (Fig. 20.5.3) to provide soft error detection. Parity generation logic is embedded locally inside each register. An XOR tree (Fig. 20.5.4) performs parity computation on the 82b (65b for integer) register data. The final parity and parity valid bits are delivered to latches for parity comparison. Parity computation takes 4 (3 for integer) clock cycles to finish. A write operation or a thread switch event triggers control logic to clear the ParityComp signal and starts the parity computation for all registers with invalid parity. The StoredParity signal gets updated after 4 (or 3) clock cycles when the new parity becomes valid. Thus, frequently written registers get less parity protection due to the latency in computing parity. However, these registers get updated frequently and thus are less susceptible to soft errors. The XOR tree remains active and computes the parity signal constantly. Thus, a parity upset caused by soft errors will set the Parity Error bit, which is available along with the register data by read operations. To facilitate post-silicon testing, a parity seed signal, ParitySeed, can be folded into the parity computation and deliberately generate a parity error. Both threads are protected by the parity checking scheme and share the same XOR tree. The parity computation and control logic introduce about 10% area overhead. An alternate implementation employing external parity generation was found to incur larger design overheads and fail timing requirements.

Figure 20.5.5 illustrates the RF wordline decoder circuit. In the previous design [2], the WRITEH signal was shared by all 128 registers, resulting in unnecessary power dissipation. The current design removes 90% of this power overhead by decoding wordline pulses to generate one WRITEH signal for each register. The writeen signal is generated using the un-clocked decoder results qualified by NCK and delivered to each register. The double-pulsed clock, PCK2, evaluates the read/write addresses on different clock phases, achieving same-cycle read and write with a single wire.

The RF arrays are organized in 8 banks, each having 16 registers. Local bitlines are loaded with 8 bitcells each and are two-way merged to global bitlines, forming dynamic 8-way OR logic at the local and global bitlines (Fig. 20.5.6). Overlapped clocking combined with pulsed read enables time borrowing for skew tolerance and eliminates foot-clocked nFETs for fast evaluation. Local and global bitlines are sensitive to leakage-induced noise due to their wide-fanin OR structure. Thus, the pulldown nFETs are selectively chosen to use non-minimal length devices. This reduces leakage current by up to 3× with minor impact on read speed. The reduction in leakage also enables smaller keepers that reduce the contention during read evaluation. Single-ended write and read scheme helps to deliver a high-density layout and improves timing over sense-amp designs [3].

The FRF and IRF are implemented in 90nm 1.1V bulk CMOS with 7-layer copper interconnect and dual-Vt FETs. The overall area is 1.29mm^2 for the FRF and 1.37mm^2 for the IRF. Reliable register read and write operations are demonstrated at various supply voltages and frequencies. At 1.1V and 2.0+GHz, the register files have a bandwidth of over 320GB/s with a power consumption of about 400mW for typical applications. Figure 20.5.7 shows the die micrograph.

Acknowledgments:
The authors wish to thanks B. Werst, M. Gibson, S. Dussinger, D. Dahle, P. Winterrowd, M. Bell, and B. Larson for their contributions.

References:
[1] S. Naffziger et al., "The Implementation of a 2-Core, Multi-Threaded Itanium®-Family Processor," *ISSCC Dig. Tech. Papers,* Paper 10.1, pp. 182-183, Feb., 2005.
[2] E. Fetzer et al., "A Fully Bypassed Six-Issue Integer Datapath and Register File on the Itanium-2 Microprocessor," *IEEE J. Solid-State Circuits*, pp. 1433-1440, Nov., 2002.
[3] M. Golden et al., "A 500MHz, Write Bypassed, 88 Entry, 90-bit Register File," *Symp. on VLSI Circuits*, pp. 105-108, June, 1999.

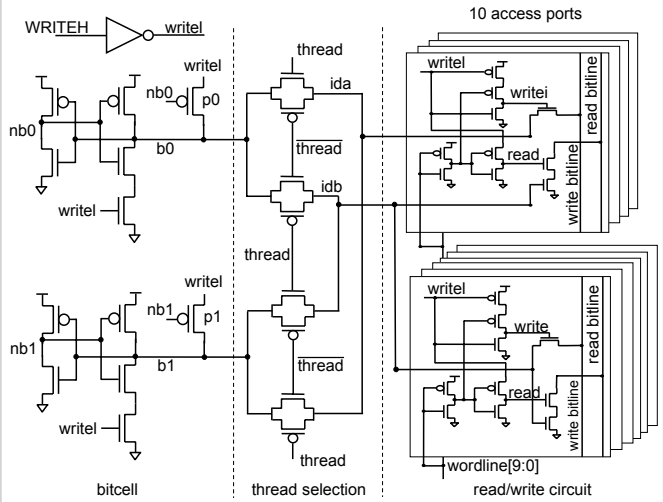

Figure 20.5.1: Register file cell.

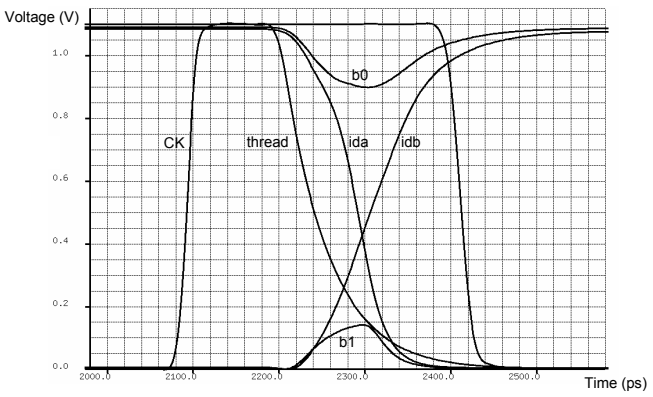

Figure 20.5.2: Waveform of thread switch.

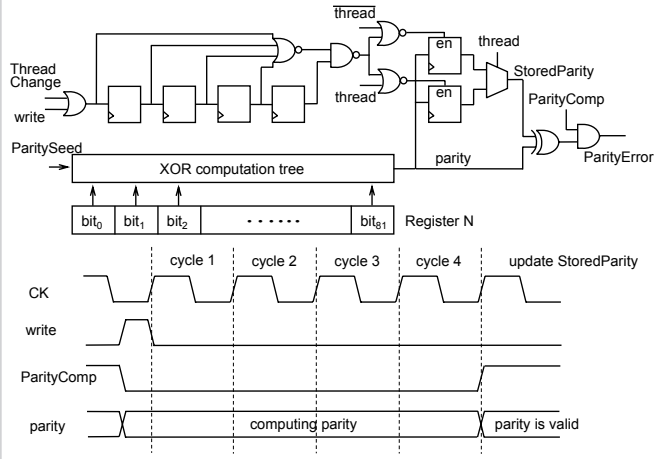

Figure 20.5.3: Parity-generation logic and timing.

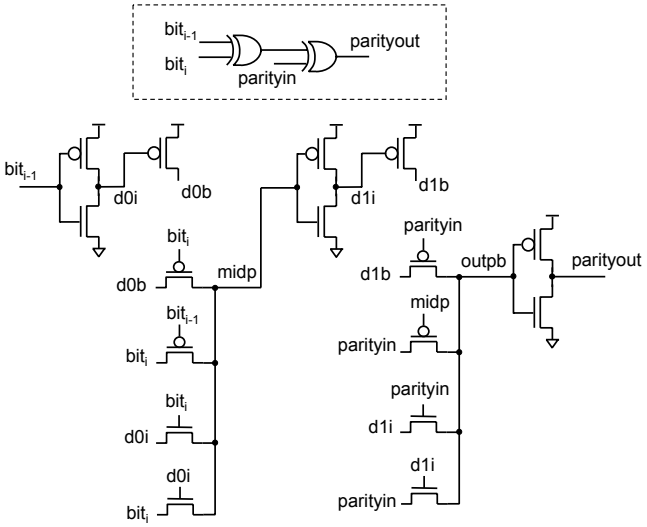

Figure 20.5.4: Parity-computation circuit.

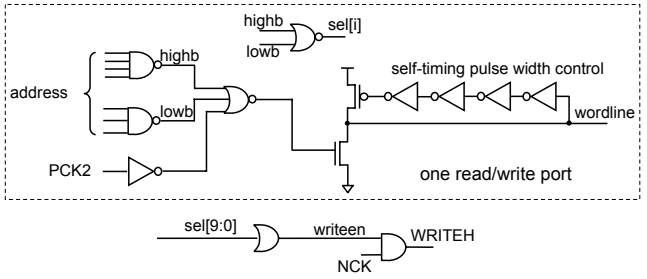

Figure 20.5.5: RF wordline decoder circuit.

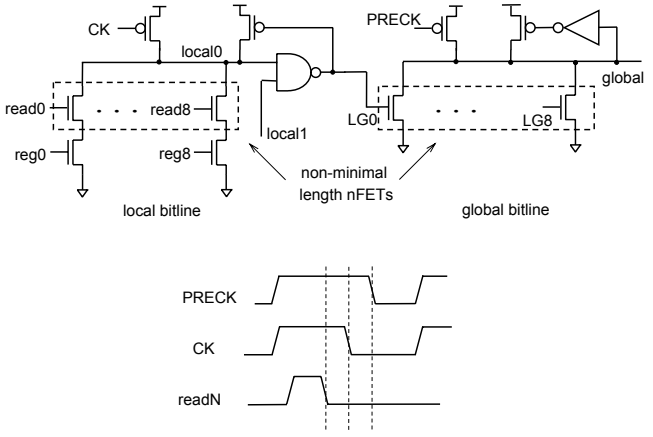

Figure 20.5.6: Bitline structure and clocking.

Continued on Page 605

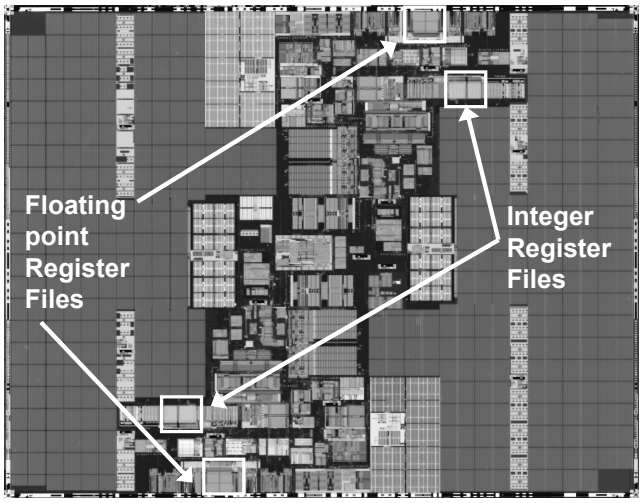

Figure 20.5.7: Die micrograph.